AD-A093 640 MARYLAND UNIV COLLEGE PARK DEPT OF PHYSICS AND ASTRONOMY F/G 12/1
COMPUTER-AIDED CLOSURE OF THE LIE ALGEBRA ASSOCIATED WITH A NON--ETC(U)
OCT 80 A REIMAN DE-AC05-79ET-53044
UNCLASSIFIED PUB-81-092 NL

1 OF 1
AD A
093640

END
DATE
FILMED
2-81
DTIC

Plasma Preprint PL #81-020

LEVEL II

AD A093640

COMPUTER-AIDED CLOSURE OF THE LIE ALGEBRA ASSOCIATED WITH A NONLINEAR PDE

by

Allan Reiman

PUB-

Physics Publication Number 81-092, PL-81-020

October 1980 21

Contract DE-AC05-79ET-53044
Department of Energy

UNIVERSITY OF MARYLAND
DEPARTMENT OF PHYSICS AND ASTRONOMY
COLLEGE PARK, MARYLAND

Computer-Aided Closure of the

Lie Algebra Associated with a Nonlinear PDE

Allan Reiman
Department of Physics and Astronomy
University of Maryland
College Park, MD 20742

I. Introduction

Since the solution of the KdV equation by Gardner, Greene, Kruskal and Miura[1] in 1967, the inverse scattering method which they discovered has been extended to solve a large class of nonlinear PDE's. Almost all such solutions have been discovered either by inspired guesswork or by starting with a scattering problem and working backwards to see what PDE's it could be used to solve. However, a method for seeking an inverse scattering solution to a given equation has emerged from the work of Wahlquist and Estabrook.[2] They have studied integrable nonlinear PDE's in terms of differential forms. Their method has been reformulated in purely classical terms by Corones[3] and further developed as a useful computational tool by Corones[3], Kaup[4], and Newell[5].

The Wahlquist-Estabrook (WE) method systematically associates a set of Lie brackets with a given nonlinear PDE. If this Lie algebra can be closed, consistent with the Jacobi identity, then an associated linear representation gives the inverse scattering transform, when one exists. Even when an inverse scattering transform does not exist, if the Lie algebra can be closed the associated linear representation can provide useful information about the solutions of the nonlinear PDE. Indeed, Kaup has shown[4] that a WE analysis of Burger's equation leads to the solution by a Cole-Hopf transformation.

By far the most laborious task in applying the WE method to a complicated nonlinear PDE is that of closing the Lie algebra.

This task involves repeated application of the Jacobi identities, which in turn requires the evaluation of a large number of Lie brackets which must be looked up in a table. The simple but repetitive nature of this task suggests that we attempt to mechanize it using a computer. However, we must also require that our program automatically perform algebraic simplifications at each step: properly distributing the Lie bracket over a sum of terms, combining like terms, multiplying numerical coefficients, etc. We have therefore used the symbolic manipulation language MACSYMA to implement our algorithms. The language has built-in capabilities for performing the standard simple algebraic manipulations from which we build up our algorithms. We also make use of MACSYMA's built-in program for solving simultaneous linear equations algebraically.

Closing the Lie algebra ordinarily involves some amount of guesswork. To effectively use the computer we must first develop a well-defined algorithm which mechanizes as much of the task as possible. We are willing to pay for this increase in mechanization by increasing the amount of algebra that must be performed. However, we also require that our implementation be sufficiently flexible to allow us to make some use of the insight we get as our algorithm progresses. We describe, in this paper, our development and implementation of an algorithm which serves these purposes. We also present an example of its use to find previously unknown solutions to a phsyically interesting set of coupled PDE's.

The PDE's which we take for our example are known in plasma physics as the wave kinetic equations (or weak turbulence three-wave equations):

$$\left(\frac{\partial}{\partial t} + v_1 \frac{\partial}{\partial x}\right) N_1 = \frac{1}{|v_2 - v_3|} (N_1N_2 - N_1N_3 + N_2N_3), \tag{1a}$$

$$\left(\frac{\partial}{\partial t} + v_2 \frac{\partial}{\partial x}\right) N_2 = \frac{1}{|v_1 - v_3|} (N_1N_3 - N_1N_2 - N_2N_3), \tag{1b}$$

$$\left(\frac{\partial}{\partial t} + v_3 \frac{\partial}{\partial x}\right) N_3 = \frac{1}{|v_2 - v_1|} (N_1N_2 - N_1N_3 + N_2N_3). \tag{1c}$$

They describe the interaction of three wave packets which are slowly varying on the space and time scales of the corresponding wavelengths and periods, yet sufficiently broad in Fourier space to justify the use of a random phase approximation. In Eqs. (1) the N's are the mean squared amplitudes of the waves and the v_j's are the group velocities. The solutions to Eqs. (1) ignoring spatial derivatives (time-only) are given in standard plasma physics texts[6]. The time-only equations are widely used to describe wave-wave interactions in a turbulent plasma. A more complete description of our results concerning Eqs. (1) is given elsewhere[7].

II. The Wahlquist-Estabrook (WE) Method

We give a brief outline of the WE method here, sufficient for explaining the role of Macsyma in our work. The workings of the method should become somewhat clearer when we go through the example in Section V. A more detailed description of the WE method is given in the references cited in the introduction.

The basic idea of the method is to imbed a given nonlinear PDE,

$$u_t = F(u,u_x,u_{xx},\ldots), \tag{2}$$

as the consistency condition for a pair of linear equations:

$$\underset{\sim}{\phi}_x = P(u,u_x,u_{xx},\ldots)\,\underset{\sim}{\phi}, \tag{3a}$$

$$\underset{\sim}{\phi}_t = Q(u,u_x,\ldots)\,\underset{\sim}{\phi}. \tag{3b}$$

Here $\underset{\sim}{\phi}$ is an auxiliary vector function of x and t, and P and Q are matrix functions of u and a finite number of its derivatives. The condition for consistency of Eqs. (3a) and (3b) is

$$\underset{\sim}{\phi}_{xt} - \underset{\sim}{\phi}_{tx} = 0, \tag{4}$$

or

$$P_t - Q_x + PQ - QP = 0. \tag{5}$$

If P and Q are given functions of u, (5) is a PDE (in general nonlinear) for u. We would like to choose the matrix functions P and Q so that Eq. (5) is the same as Eq. (2).

Applying the chain rule to Eq. (5), using Eq. (2) to replace u_t by F, and defining

$$[P,Q] \equiv PQ - QP \tag{6}$$

we obtain

$$P_u F + P_{u_x} u_{xt} + \dots - Q_u u_x - Q_{u_x} u_{xx} - \dots + [P,Q] = 0. \tag{7}$$

Note that the highest derivative of u appearing explicitly in (7) is not an argument of P or Q. Since the initial value of u is arbitrary, we therefore demand that the coefficient of this term vanish. Solving the resulting equation, we substitute back into (7), and then we iterate this process. It will be clearer how this works when the example is presented in section V. When this procedure can be completed we are led to a solution for P and Q of the form

$$P = \sum_{j=1}^{\ell} X_j(u, u_x, \dots), \tag{8a}$$

$$Q = \sum_{j=1}^{m} Y_j(u, u_x, \dots), \tag{8b}$$

where various of the commutators between X_j's and Y_j's are determined.

If we can close the "unclosed Lie algebra" determined by the X_j's, the Y_j's, and the commutation relations we have found, then we can find a linear representation for the Lie algebra. This linear representation then gives us a solution for the matrices P and Q. For those nonlinear equations which are soluble by means of an inverse scattering method, a free parameter appears in this closure. This free parameter corresponds to the eigenvalue. For nonlinear equations soluble by a Cole-Hopf type of transformation, no such free parameter appears.

Closing the Lie algebra is a critical step, often involving much guesswork and much laborious algebra. Systematic procedures that work for relatively simple equations like the KdV equation become discouragingly laborious when applied to the Lie algebras for more complicated equations. We therefore use Macsyma to perform the algebra involved in this step. Two algorithms which we have implemented on Macsyma for this purpose are described in the next section. Taken together, the two algorithms appear to provide a very general method of closing the algebra.

We should mention here the implementation by Wahlquist and Estabrook of the method of differential forms on MACSYMA.[8] While their concern has been to bring to bear the full power of the formalism, we use only the properties of the Lie bracket. We have gone beyond their work in developing and implementing the set of algorithms described in this paper for closing an "incomplete" Lie algebra.

III. Algorithms for Closing the Algebra

Systematic procedures for closing the Lie algebra involve at each step the evaluation of the Jacobi identity

$$[A,[B,C]] + [B,[C,A]] + [C,[A,B]] = 0 \tag{9}$$

for all triplets A,B,C such that the commutator is known for each pair of the triplet. The identities obtained may determine previously unknown brackets. If they do, then the new brackets are substituted back into the Jacobi identities. The process continues until no more new commutators are determined. Not only is this an extremely tedious process if there are a large number of elements, but it is also a process very sensitive to error. Algebraic errors generally lead, after considerable wasted effort, to the possibly false conclusion that the Lie algebra must be trivial.

One procedure which we have usefully implemented on Macsyma adds new elements, defined equal to some of the unknown commutators. We add one element at a time, substituting all newly determined commutators into the Jacobi identities as we proceed. For the example which we will describe in section V this algorithm led to a closed Lie algebra. Furthermore, when this algorithm closes the Lie algebra it allows us to determine whether a free parameter such as that associated with the eigenvalue of an inverse scattering transform exists.

For those sets of commutators which are not consistent with any nonabelian closed Lie algebra, the algorithm described usually leads quickly to an Abelian algebra. An example of this is the WE analysis for the combined Burgers-KdV equation

$$u_t + \alpha u_{xxx} + \beta u_{xx} + \gamma u_x u = 0. \tag{10}$$

Taking

$$\underset{\sim}{\phi}_x = P(u)\underset{\sim}{\phi},$$

$$\underset{\sim}{\phi}_t = Q(u,u_x,u_{xx})\ \underset{\sim}{\phi},$$

the WE analysis leads systematically to a set of commutation relations. Applying the algorithm described, we are led to the conclusion that all elements of the Lie algebra must commute. This calculation would have been prohibitively laborious if it had not been done on Macsyma.

The algorithm we have described gives a useful first attempt at closing the Lie algebra. For the example we present in section V it is all that is necessary. But the algorithm does not always terminate. For the KdV equation,

$$u_t + u_{xxx} + 6uu_x = 0, \tag{11}$$

the WE ansatz

$$\underset{\sim}{\phi}_x = P(u)\underset{\sim}{\phi}, \ \underset{\sim}{\phi}_t = Q(u,u_x,u_{xx})\underset{\sim}{\phi} \tag{12}$$

leads to

$$P = X_1 + uX_2 \tag{13a}$$

$$Q = - u_{xx}X_2 - u_xX_3 - 3u^2X_2 - \frac{u^2}{2} X_5 - uX_4 + \overline{X} \tag{13b}$$

where

$$[X_1,X_2] = X_3, \ [X_1,X_3] = X_4, \ [X_2,X_3] = X_5$$

$$[X_2,X_5] = [X_1,\overline{X}] = 0, \ [X_1,X_4] = [X_2,\overline{X}] \tag{14}$$

These brackets may be imbedded in arbitrarily large Lie algebras. The algebra may, however, be closed without adding any new elements. Set

$$\overline{X} = C_1X_1 + C_2X_2 + \dots + C_5X_5, \tag{15}$$

where the C_j's are constants to be determined. By evaluating all Jacobi identities we determine all but one of these constants and all

remaining commutators. One constant remains arbitrary, corresponding to the eigenvalue in the associated inverse scattering transform. The ansatz (15) is suggested by the fact[5] that as $x \to \pm\infty$, $Q \to \bar{X}$, so that $\bar{X}$ must contain the eigenvalue if we are to solve the nonlinear PDE by an inverse scattering method. This second algorithm appears to be a very general one for finding the inverse scattering transform of an integrable equation.

IV. Implementation on Macsyma

Macsyma has a built-in noncommutative product operator whose purpose is to represent multiplication of matrices. For our purposes, we find it convenient to use this noncommutative product to represent the Lie bracket. Having used a built-in MACSYMA operator, we can apply various operations available in MACSYMA for evaluation and simplification of algebraic expressions. To guarantee that the Lie bracket will be handled properly by these operations, we reset several of the MACSYMA flags associated with the noncommutative product. By doing so we give this product all the desired properties except antisymmetry, which must be built into the table of commutators.

The noncommutative product operator is representated in MACSYMA by a dot, and so is called the "dot operator." The double bracket which arises in the Jacobi identities

$$[A,[B,C]] \ ,$$

is simply represented as

$$A.(B.C) \ .$$

The dot operator is normally taken to be associative in MACSYMA. By setting the DOTASSOC flag to FALSE we make it nonassociative, as required by the Lie brackets. Similarly, by setting the DOTDISTRIB flag to TRUE we allow A.(B+C) to simplify to A.B + A.C.

To distinguish between those variables which are elements of our Lie

algabra and those which represent numerical coefficients, MACSYMA allows us to declare variables to be nonscalar. This is done using the function DECLARE. The way in which scalars are handled is then controlled by the flag DOTSCRULES. We set DOTSCRULES to TRUE. If c is a scalar, c.A or A.c will then simplify to c*A, and A.(c*B) will simplify to c*(A.B). With DOTCONSTRULES set to TRUE, purely numerical coefficients are treated in the same manner.

The table of commutators is stored as a list of equations, with each equation of the form

$$A.B = C \;. \tag{16}$$

This allows us to use the EV command, which will evaluate a given expression subject to a list of equations. For example, EV(A.B + D,TABLE) yields the expression C+D if TABLE is a list of equations including Eq. (16). If EXPR is a more complicated expression containing Lie brackets, then the command EV(EXPR,TABLE) will substitute C for A.B each time the latter appears in EXPR. By adding the argument INFEVAL, we use EV in an "infinite evaluation" mode. If D.C = E is also determined by TABLE, and D.(A.B) appears in EXPR, EV(EXPR,TABLE,INFEVAL) substitutes an E. Note, however, that EV does not distribute over the dot product. If A.B = E+F, then only the first layer of D.(A.B) can be evaluated by an EV command. To evaluate further we must first use the EXPAND command. This command yields a simplified expression in which the dot and scalar products have been distributed over sums of terms, scalar multiplication has been factored out of the dot product, etc.

The program reads in the number of initial elements (N) in the unclosed algebra, and then reads in the known structure constants of the Lie algebra, using the READ command in a loop. Of course at most $N(N-1)/2$ such constants must be given, with the antisymmetric twin of each equation automatically added to the table, and with A.A automatically set to zero. Given the structure constants, the program proceeds to evaluate all Jacobi identities, using two layers of EXPAND and EV commands for each.

To implement the algorithms described in the previous section, the program defines functions which: evaluate a given Jacobi identity; update all Jacobi identities using new information; update a given Jacobi identity; add a given equation to the table; evaluate all Jacobi identities involving a newly added element of the algebra; store the updated table along with all defined functions, options, and properties on disk; display the table of structure constants. These functions perform the laborious algebraic tasks involved in trying to close the Lie algebra. They allow the user to implement the algorithms of the previous section in a flexible fashion, making use of any new insight as he gains experience in attempting to close such an algebra.

V. An Example: The Kinetic Three-Wave Equations

We have used Macsyma to close the Lie algebra for Eqs. (1). Here we first derive the appropriate commutation relations, and then describe our use of Macsyma to close the algebra.

Eqs. (3) are our starting point, with P and Q each assumed to depend only on the three N_j's. Setting the coefficient of N_{jx} equal to zero for each j, we obtain

$$v_j P_{N_j} = Q_{N_j} \text{ , } j = 1,2,3. \tag{17}$$

Taking cross derivatives of these equations, we find

$$v_j P_{N_j N_k} = v_k P_{N_k N_j} . \tag{18}$$

Since the group velocities are assumed all different, Eq. (18) implies

$$P_{N_j N_k} = 0 \tag{19}$$

or

$$P = \sum_{j=1}^{3} g_j(N_j) + Z, \tag{20a}$$

$$Q = - \sum_{j=1}^{3} v_j g_j(N_j) + W. \tag{20b}$$

From Eq. (7) we further find that $g_j''' = 0$, so that g_j must be of the form

$$g_j = N_j X_j + N_j^2 V_j,$$

where we have absorbed the possible constant term into W and Z. The Jacobi identities imply that each V_j must commute with all elements of the algebra. So the V_j's can be set to zero without changing the structure of the algebra.

We have concluded that P and Q must be of the form

$$P = \sum_{j=1}^{3} X_j N_j + Z, \tag{21a}$$

$$Q = -\sum_{j=1}^{3} V_j X_j N_j + W. \tag{21b}$$

Substitute these expressions back into Eq. (7), setting the coefficient of each term in the resulting polynomial in the N_j's to zero, to obtain a set of commutation relations. To simplify the expression of these relations we define

$$Y_1 = X_1,\ Y_2 = -X_2,\ Y_3 = X_3. \tag{22}$$

In terms of the Y_j's, our commutation relations are

$$(v_m - v_n)[Y_m, Y_n] = \frac{1}{|v_2 - v_3|} Y_1 + \frac{1}{|v_3 - v_1|} Y_2 + \frac{1}{|v_2 - v_1|} Y_3,$$

$$m \neq n$$

$$[Y_m,W] = v_m[Z,Y_m], \tag{23}$$

$$[W,Z] = 0.$$

The commutator of Y_m with Z or W is not determined.

Now we apply the first algorithm described in section III, using Macsyma. Let

$$[Z,Y_j] = U_j.$$

Evaluating all Jacobi identities, we find that the commutators of U_j with W and Z are now left undetermined. Let

$$[Z,U_j] = \bar{U}_j.$$

Evaluating all Jacobi identities we find that we must again extend the algebra. Let

$$[Z,\bar{U}_j] = \tilde{U}_j.$$

Now when we evaluate the Jacobi identities the algebra collapses back down. We find that $\bar{U}_j = 0$ for all j. We obtain relations between various of the elements. Substituting back into the Jacobi identities, we are able to determine all commutators.

We reduce the number of elements in our Lie algebra by setting to zero all elements or combinations of elements which commute with every other element. This does not affect the commutation relations, but does simplify our search for a linear representation.

As a final simplification, we work in a reference frame where some of the structure constants vanish. We choose our reference frame so that

$$v_1|v_3 - v_2| + v_2|v_1 - v_3| + v_3|v_2 - v_1| = 0. \tag{24}$$

Since we have assumed that all three group velocities are different, we can always find such a frame. In this reference frame we can close the Lie algebra by setting

$$[Y_1,Z] = -\frac{\mathrm{sgn}(v_2 - v_1) + \mathrm{sgn}(v_3 - v_1)}{v_1(v_3 - v_1)(v_1 - v_2)} W , \tag{25a}$$

$$[Y_3,Z] = \frac{\mathrm{sgn}(v_3 - v_2) + \mathrm{sgn}(v_3 - v_1)}{v_3(v_3 - v_2)(v_1 - v_3)} W , \tag{25b}$$

$$Y_2 = (v_1 - v_3)(\frac{1}{v_2 - v_3} Y_1 + \frac{1}{v_1 - v_2} Y_3). \tag{25c}$$

These relations, along with Eqs. (23), define a closed Lie algebra.

The final result looks deceptively simple. It probably would have taken a good deal of guessing and playing around with the algebra to arrive at Eqs. (24) and (25) without the systematic approach

we used. And the systematic approach would have been discouragingly laborious without the use of Macsyma. Furthermore, having gone through our systematic procedure, we are now assured that the algebra we obtained is the largest possible. This is of interest because it implies that there can be no free parameters of the sort that arises when a nonlinear PDE can be solved by an inverse scattering method.

Using the closed Lie algebra thus obtained, we have found a linear 3 x 3 representation, and have thus been able to imbed the nonlinear Eqs. (1) in the linear Eqs. (3). We have been able to use this imbedding to obtain an interesting set of exact multi-shock solutions. Except for the single shock solutions[8] these solutions are new. They are described elsewhere[7].

Acknowledgment

The author is indebted to D. Kaup and A. Newell for many valuable discussions.

* Work done under an ONR grant.

References

1. C. Gardner, J. Greene, M. Kruskal, and R. Miura, Phys. Rev. Letters 19, 1095 (1967).
2. H. D. Wahlquist and P. B. Estabrook, J. Math. Phys. 16, 1 (1975); ibid. 17, 1293 (1976).
3. J. Corones, J. Math. Phys. 17, 756 (1976).
4. D. Kaup, MIT Plasma Research Report, PRR 79/6 (1979).
5. A. Newell, Proc. R. Soc. London 365, 283 (1979).
6. R. Davidson, Methods in Nonlinear Plasma Theory, Academic Press, 1972; V. N. Tsytovich, Nonlinear Effects in Plasma, Plenum Press, N.Y., 1970.
7. A. Reiman and D. Kaup, Bull. Am. Phys. Soc. 23, 795; Phys. Fluids Feb. 1981 (to appear).
8. H. D. Wahlquist, Proceedings of the First MACSYMA User's Conference, p. 71 (1978).
9. Ohsawa and Tanuiti, J. Phys. Soc. Jap. 35, 907; ibid. 36, 313.